

Petre Rău

**LECTȚII
DE
ȘAH – COMPUTER**

Editura InfoRapArt

C U P R I N S

În loc de prefață	3
Concepte de bază în șahul programat	7
1. Reprezentarea tablei de șah	8
2. Codificarea pieselor	10
3. Direcții de căutare pentru efectuarea unei mutări	12
4. Determinarea mutărilor posibile	15
5. Codificarea mutărilor	17
6. Generatorul de mutări	19
7. Generatorul de mutări. Nivelul de joc	21
8. Funcția de evaluare	23
9. Alegerea mutării optime	26
10. Tratarea excepțiilor în șahul programat	28
11. Fișiere de deschidere și finaluri	31
12.Considerații finale	32
Aspecte geometrice ale tablei de șah în programarea finalurilor	35
Funcția de evaluare în șahul programat	40

În loc de prefață

(Extras dintr-un interviu la Radio)

Reporter: - Ce reprezintă calculatorul pentru dumneavoastră?

P.R.: - Calculatorul reprezintă unealta cu care omul poate atinge mai repede perfecțiunea în fața naturii. El este un instrument al cunoașterii, menit să-l apropie pe om de adevăr, de esență.

Reporter: - Mai concret, la ce vă folosește calculatorul?

P.R.: - În ceea ce mă privește calculatorul mi-a fost mereu de mare ajutor și asta nu numai pentru că profesia mea este legată de acesta, ci și datorită faptului că aproape de fiecare dată când m-am aflat la limita imposibilității de a rezolva o problemă i-am cerut sprijinul. Desigur, nu mă refer numai la ceea ce poate oferi calculatorul vizavi de societate, producție, de economie în general, ci și la ajutorul oferit cercetătorului, în domeniul creativității etc. Este nu numai util, dar și minunat să poți mânui asemenea instrumente.

Reporter: - Din câte știm ați experimentat multe lucruri pe calculator. Puteți oferi mai multe amănunte în acest sens?

P.R.: - Da. În general m-au preocupat foarte mult problemele pe care le ridică astăzi inteligența artificială. La noi în țară începuturile acestora au fost foarte dificile, ele au vizat mai mult planul teoretic al fenomenului și mai puțin realizările practice. Poate și din acest motiv inteligența artificială nu și-a putut crea un drum sigur și nu s-a bucurat de popularitatea pe care ar fi meritat-o. Părerea mea este că, fiecare cercetător din acest domeniu ar trebui, mai devreme sau mai târziu, să experimenteze în mod practic câte ceva, de pildă să vadă nu numai teoretic ce înseamnă un program autoinstruibil, sau un demonstrator de teoreme, atunci când sunt puse la lucru, ce înseamnă un motor de inferențe și să experimenteze un motor de inferențe logice, ce dificultăți apar atunci când încerci să-i ceri calculatorului să comunice în limbaj natural sau să se supună unui limbaj logic în esența lui. Cercetătorul poate sesiza mai bine în acest mod adevăratele probleme care se nasc atunci când încerci să-l faci pe calculator să se "comporte" la fel ca o inteligență umană. Este convenabil și, dacă vreți, prea pașnic și comod să te mărginești la a sesiza nuanțe ale comportamentului unei inteligențe umane, mai greu este însă găsirea unor „rețete” pentru implementarea acestora pe un calculator.

Reporter: - Și dumneavoastră ați găsit? Ați experimentat ceva din toate acestea?

P.R. - Eu am pornit tocmai de la această ultimă idee și anume aceea de a experimenta cât mai multe similitudini de exprimare inteligentă a unui program de calculator în fața inteligenței umane. Un mare cobai al inteligenței artificiale a fost, este și va mai rămâne încă jocul de șah...

Reporter: - Acesta a fost începutul?

P.R. - Da.

Reporter: - De ce tocmai cu jocul de șah?

P.R.: - Foarte simplu! Pentru acest joc planeta noastră a investit în ultimile două secole poate cea mai mare cantitate de inteligență creatoare și astăzi tot mai putem spune că tainele lui n-au fost pe deplin clarificate. A trebuit, periodic, să intervină minți aproape diabolice cum au fost toți marii campioni pentru a mai avansa câțiva pași în cunoașterea acestui joc. Va trebui deci să treacă poate alte secole pentru ca el să devină în totalitate cunoscut minții omenești. Singura salvare rămâne totuși calculatorul, deși el, până astăzi, n-a atins nici pe departe cele mai importante cote ale performanței umane.

Reporter: - Considerați că ceea ce n-a putut face mintea omenească ar putea face calculatorul?

P.R.: - Nu tocmai, deși, ținând cont de finitudinea jocului și de logica înșiruirii operațiunilor pe tabla de șah calculatorul, dacă ar fi suficient de înzestrat, ar trebui să-l trateze în întregime. Dar deocamdată el se află în dificultate în fața imensității de variante ale jocului propriu zis. Și atunci, până când se va naște un calculator care să vadă finalul jocului încă de la primele mutări, ceea ce eu cred că se va petrece nu peste mult timp, calculatorul este învățat doar să imite ceea ce gândește jucătorul uman în fața tablei de șah și, eventual, să egaleze performanțele acestuia.

Reporter: - Ar trebui să înțelegem că dumneavoastră v-ați ocupat de această ultimă problemă?

P.R.: - Da. M-am ocupat nu să implementez niște strategii particulare de joc, cele pe care le aplică fiecare jucător avansat și care au fost "copiate" aproape identic în mai toate programele de șah existente astăzi în lume, ci să implementez ceea ce jucătorul singur nu ar putea face cu deplină facilități în fața tablei de șah. Mai precis am pornit de la ideea că nu anumite strategii de joc aduc câștigul, ci mai degrabă înțelegerea, în ansamblul lor, a tuturor operațiunilor logice care se execută pe tabla de șah și corelarea acestora.

Reporter: - Și cum ați concretizat această idee?

P.R.: - În primul rând aş adăuga că eu am lansat ideea de dominare și de utilitate în joc. Cu fiecare mutare făcută, calculatorul trebuie să rețină ca unic criteriu de eficiență ideea de dominare caracterizată în principal de totalitatea operațiunilor de atac, de apărare și de control al pieselor și câmpurilor de joc. Din punct de vedere matematic, toate acestea sunt înglobate în ceea ce se cheamă funcția de evaluare, care este o funcție mai mult sau mai puțin liniară, alimentată de un sistem de constante prestabilite și de condițiile concrete care intervin pe tabla de șah. Totul depinde de stabilirea acestor constante care exprimă, dacă vreți, la modul cel mai simplu, valoarea relației de atac, apărare sau control. Această matrice a constantelor poate fi permanent îmbunătățită, ba mai mult, ea poate fi supusă unui riguros sistem de adaptare într-un program autoinstruibil.

Reporter: - Vrem să putem înțelege diferența dintre ideea de care vorbiți și celelalte utilizate până în prezent în lumea șahului programat.

P.R.: - Să vă dau un exemplu. Porniți, dacă vreți, de la un joc oarecare cunoscut, care presupune doi parteneri de întrecere. Dacă există o singură strategie de câștig, atunci lucrurile sunt clare, câștigul fiind asigurat celui care o stăpânește și o respectă cu strictețe. El va efectua un șir logic de operațiuni fidele acestei strategii. Dacă jocul ar avea două strategii de câștig atunci lucrurile se complică, în timpul jocului putând apărea condiții când sunt valabile ambele strategii sau numai una din ele. În acest caz, jucătorul trebuie să sesizeze totdeauna corelarea acestora, altfel câștigul nu mai poate fi sigur. Dacă vreți un exemplu mai simplu, să presupunem că doi parteneri se întrec în a străbate cel mai scurt traseu al unui labirint necunoscut. Există două strategii simple de abordat: una care presupune parcurgerea labirintului

urmând fiecare perete din dreapta, și duala sa, pentru peretele din stânga. Oricare dintre ele poate fi câștigătoare sau nu. O a treia strategie, dacă ar fi inventată, este aproape obligatoriu să îmbine componente ale celorlalte două și să coreleze părți ale acestora pentru a găsi un drum și mai scurt, mai optim...

Reporter: - Despre punerea în practică a acestora idei ce ne puteți spune?

P.R.: - Ar trebui să încep prin a vă spune că, deși astăzi lumea este invadată pur și simplu de claculatoare și programe de șah, problema algoritmizării acestui joc încă este actuală. Nicăieri în lume nu s-a putut emite pretenția finalizării acestei probleme și numeroase premii au fost și încă mai sunt oferite în acest scop. Eu am realizat mai întâi, în anul 1985, după aproape doi ani de muncă, un program de rezolvare a problemelor de șah cu care am participat la faza finală a concursului de programare "Calculatorul în sprijinul societății", organizat în principal de revista "Știință și tehnică". Programul a fost foarte apreciat la acea vreme, era primul de acest gen creat în țara noastră, dar eu personal nu am vrut să mă opresc în această fază. Dorința mea era să creez un program de jucat șah. Și parțial am reușit. La câteva luni după aceea am prezentat conceptul noului meu program la primul festival național pe această temă, susținut la Băile Herculane în martie 1986. Ideile mele s-au bucurat de o largă apreciere, programul a fost declarat ca fiind a doua concepție originală românească în acest domeniu, prima fiind semnată în 1976 de cunoscutul cercetător Viorel Darie. Imediat după aceea am început să mă bucur de sprijinul moral și documentar al unui mare animator al șahului electronic, pe atunci președintele Comisiei Centrale de Șah- Computer, Uly Vălușanu. Ideile au fost multe și frumoase, dar cine pe vremea aceea îți putea oferi condiții favorabile pentru punerea lor în practică? M-am lovit de numeroase obstacole, multe dintre ele neputându-le depăși. Trebuie să menționez că de această problemă m-am ocupat numai în timpul meu liber, lipsit de mijloace tehnice adecvate, singur, fără colaboratori, iar într-un oraș de provincie, chiar dacă este vorba de Galați, preocupările și interesul pentru această activitate au fost și mai sunt încă foarte rare. Munca este însă imensă și nu poate fi acoperită în întregime de un singur om. Oricum, rezultatele la care am ajuns mi-au oferit destule satisfacții personale și mulți consideră că am realizat pași importanți...

Reporter: - Să înțelegem, prin cele ce ne-ați mărturisit, că așteptați propuneri de colaborare sau sprijin în acest sens?

P.R.: - De ce nu? Experiența pe care am câștigat-o în acest domeniu este imensă. Realizarea unui program de șah este astăzi relativ ușoară și chiar un elev ar putea-o aborda. Programul însă, algoritmul propriu-zis, nu înseamnă nimic. El îi arată calculatorului cum să mute corect piesele pe tabla de șah, dar nu să și joace corect și bine. Problema cheie în șahul-computer rămâne aceea subrutină încă fantomatică, care este funcția de evaluare. Ori de câte ori se înlocuiește această funcție cu o alta mai bună, crește și nivelul de joc al programului.

Reporter: - Cred că ar fi util pentru cititorii noștri să le oferiți mai multe detalii în legătură cu această enigmatică funcție de evaluare. Bineînțeles, dacă nu este un secret...

P.R.: - Nu este nici un secret. Eu aș zice mai degrabă că este un fel de mister. Pentru mai multe detalii despre funcția de evaluare invit cititorii să consulte articolul meu intitulat "Funcția de evaluare în șahul programat"...

Reporter: - Sunteți un bun jucător de șah? Dețineți o clasificare sportivă?

P.R.: - Am practicat foarte mult acest joc și, desigur, îl mai practic și astăzi atunci când timpul îmi permite. N-am fost preocupat niciodată de performanțe și clasificări în acest domeniu, jocul m-a atras mai mult pentru frumusețea lui și, de ce nu, ca obiect de studio pentru cercetările mele. Totuși, am jucat multe competiții șahiste, îndeosebi șah prin corespondență, am câștigat mai multe concursuri dar nu

am depășit niciodată nivelul unui jucător de categoria I.

Reporter: - Este important să fii un foarte bun jucător pentru a face un program performant de jucat șah?

P.R.: - Părerea mea este că nu. Dar pentru mai multe explicații vă rog să consultați articolul meu intitulat "Geometria tablei de șah în programarea finalurilor" care abordează în detaliu acest subiect.

Reporter: - În final ce ați dori să adresați celor interesați de acest subiect?

P.R.: - Cred că un gând bun și un îndemn către îndreptarea atenției spre nebănuitele frumuseți și satisfacții pe care ți le oferă jocul de șah în general și șahul electronic în special. Personal cred că problema șahului-computer va putea fi depășită, ca de altfel marea majoritate a celorlalte probleme care fac obiectul de studiu al inteligenței artificiale. Mai devreme sau mai târziu omul va depăși cele mai multe dificultăți și-și va face din calculator un prieten. Un prieten care nu trădează, un prieten cu adevărat util.

Simona Șerban, 1991

CONCEPTE DE BAZĂ ÎN ȘAHUL PROGRAMAT

Materialul care urmează, structurat sub forma unor lecții care cuprind ideile și conceptele de bază utilizate în realizarea unui program de șah pentru computer, se adresează în primul rând tuturor celor interesați în acest domeniu, dar și oricărui șahist obișnuit care poate întâlni aici multe principiile generale precum și unele strategii particulare utilizate mai des în practica acestui joc.

În cea mai mare parte materialul conține idei și concepte originale folosite de autor pe parcursul a mai multor ani de experimentări în acest domeniu.

Fără îndoială, întregul arsenal de cunoștințe necesare abordării problematicei șahului-computer este deosebit de complex și vast. Autorul speră totuși ca din puținele pagini pe care le dedică aici acestui subiect să clarifice principalele noțiuni și concepte care stau la baza oricărui program de jucat șah și să deschidă unele căi de noi de cercetare care ar putea intra în atenția unor categorii de cititori interesați în definitivarea completă a acestei probleme.

Autorul s-a străduit să prezinte materialul la modul cel mai accesibil, astfel că se poate spera că cititorul interesat va putea asimila fără prea mari dificultăți ABC-ul șahului-computer, acest cobai pe care numeroase minți inteligente l-au “adoptat” fără reticențe în fruntea bătăliei de pe tărâmul pașnic al inteligenței artificiale.

1. Reprezentarea tablei de șah

În memoria calculatorului tabla de șah poate fi reprezentată, cel mai simplu și sugestiv în același timp, printr-o matrice pătratică de dimensiune 8x8. Fiecare element al acesteia, având coordonatele (i, j) , este pus în corespondență biunivocă cu un pătrat al tablei de șah care, în notație clasică, poate fi scris sub formă xy , unde x este una din literele a,b,c,d,e,f,g,h, iar y este o cifră de la 1 la 8.

Relația de transformare a coordonatelor clasice în coordonate matriceale este o relație liniară de forma

$$\begin{aligned} i &= 9 - y \\ j &= \text{ordnum}(x), \end{aligned}$$

unde prin $\text{ordnum}(x)$ înțelegem ordinul numeric al variabilei x într-o corespondență indexată a primelor opt litere ale alfabetului. În figura 1 pot fi văzute simultan cele două sisteme de notație pentru tabla de șah, sistemul clasic și cel matriceal, precum și corespondența dintre cele două tipuri de notație.

De exemplu, câmpului f5 din notația clasică îi corespunde câmpul de coordonate (4,6) din notația matriceală.

8								
7								
6								
5						f5 (4,6)		
4								
3								
2								
1								
	a	b	c	d	e	f	g	h

Fig.1 – Reprezentarea tablei de șah în computer

Există, desigur, și alte sisteme de reprezentare a tablei de șah în calculator. Cel mai des utilizat constă dintr-o tablă extinsă de 120 de câmpuri, de fapt o tablă

extinsă la 12 linii și 10 coloane. Câmpurile acestei table sunt numerotate de la 0 la 119 și conțin cele 64 de câmpuri de bază, celelalte fiind utilizate în scopul verificării unor mutări cu salturi care pot depăși marginile tablei obișnuite. Această categorie de reprezentare ale tablei de șah în memoria calculatorului precum și altele nu vor face obiectul acestor lecții; pentru cei care doresc totuși să cunoască și alte tipuri de reprezentări facem invitația de a consulta unele articole specifice din almanahul “Literatură și jocurile minții - Planeta șah” din anul 1989.

2. Codificarea pieselor

Cele șase piese distincte precum și cele două culori folosite în jocul de șah trebuie să fie foarte bine definite și determinate în șahul electronic. Ele reprezintă, de fapt și de drept, elementele de bază ale jocului și, din punct de vedere al programatorului, toate calculele, testele și selecțiile se fac în virtutea unor criterii de codificare cât mai clar exprimate și reprezentate.

De la bun început trebuie să precizăm că, cele două culori ale câmpurilor de pe tabla obișnuită de joc nu au nici o semnificație pentru computer, așa cum de altfel nici pentru un jucător oarecare, “neînrobit” încă de scheme și tactici bazate pe astfel de criterii. Se pare că la originea sa jocul de șah nici nu cuprindea cele două culori ale câmpurilor tablei de joc. Acestea au fost introduse mult mai târziu, ca un veritabil amendament pentru gândirea schematică și vizuală. De aceea, așa cum se va putea constata, noi nu vom folosi niciodată noțiunile de câmpuri albe și negre în tot ceea ce ne interesează.

Cea mai obișnuită codificare a pieselor de șah este cea algebrică. Ea constă din utilizarea unor coduri numerice pentru cele șase piese și a semnelor algebrice + (plus) și - (minus) pentru culorile acestora. Astfel, codurile utilizate pot fi:

- 1 pentru **Rege**
- 2 pentru **Damă**
- 3 pentru **Turn**
- 4 pentru **Nebun**
- 5 pentru **Cal**
- 6 pentru **Pion**.

Ele vor purta totdeauna în față semnele algebrice de care vorbeam mai înainte și anume: + (plus) pentru o piesă de culoare albă, și - (minus) pentru o piesă de culoare neagră. De exemplu, un cal alb va avea codul +5, iar un rege negru va avea codul -1.

Acest sistem de codificare oferă un mare avantaj și anume acela că astfel de coduri pot fi ușor manipulate în calcule, comparări și evaluări, dar mai ales pentru ceea ce se cheamă în programare problema indexării, adică a accesului și localizării informației pe baza unei ordonări prestabilite.

O semnificație aparte o are codul **0** (zero), el reprezentând “conținutul” unui câmp liber. În figura 2 se poate vedea tabla de șah pregătită în memoria calculatorului la începutul unei partide, adică atunci când piesele se află în poziția lor inițială.

8	-3	-5	-4	-2	-1	-4	-5	-3
7	-6	-6	-6	-6	-6	-6	-6	-6
6	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
2	+6	+6	+6	+6	+6	+6	+6	+6
1	+3	+5	+4	+2	+1	+4	+5	+3
	a	b	c	d	e	f	g	h

Fig.2 – Tabla de șah cu poziția inițială a pieselor

În acest moment deja ne-am înarmat cu câteva instrumente pentru a putea efectua un control asupra conținutului unei table de șah. De exemplu, verificând conținutul tablei de șah care este încărcată cu poziția inițială, în câmpul de coordonate (4, 6), vom găsi valoarea 0 (zero) și vom ști că avem de-a face cu un câmp liber, adică fără nici o piesă de culoare albă sau neagră. Sau, verificând câmpul de coordonate (7,5), vom găsi valoarea +6 și vom ști astfel că acolo se află un pion de culoare albă.

Fără îndoială, sistemul de codificare prezentat mai sus nu este unic. Există multe alte posibilități de codificare a pieselor și câmpurilor. Toate însă au la bază criterii de limitare a numărului de instrucțiuni folosite în program pentru a efectua calcule, comparări și selectări de soluții. Însuși limbajul de programare utilizat poate impune alegerea și utilizarea unui sistem de codificare adecvat.

3. Direcții de căutare pentru efectuarea unei mutări

În această lecție vom încerca să prezentăm posibilitățile de mișcare a pieselor pe o tablă de șah codificată așa cum am stabilit în lecția anterioară.

Fără îndoială, mișcarea pieselor pe tabla de șah, fie ea și cea simulată pe un calculator, trebuie să se facă cu respectarea tuturor regulilor cunoscute ale jocului propriu-zis. De aceea este util să începem prin a sintetiza principiile de bază ale acestor mișcări supuse unor reguli prestabilite.

Fie (i_0, j_0) coordonatele unui câmp al tablei de șah pe care se află o piesă care trebuie mutată.

O mutare nu este posibilă, adică nu poate fi efectuată, în următoarele situații:

- câmpul de destinație este ocupat de o piesă de aceeași culoare,
- regele propriu rămâne în șah (prin descoperire sau prin mutarea acestuia),
- este o rocadă care nu mai poate fi efectuată,
- este o mutare “en-passent” care nu mai poate fi efectuată.

Câte mutări pot fi efectuate cu piesa aflată pe câmpul inițial (i_0, j_0) ? Există piese care se pot deplasa pe linii, coloane sau diagonale, peste unul sau mai multe câmpuri deodată și, desigur, pentru acestea trebuie să prevedem posibilitățile când mutările pot fi făcute în spiritul regulamentului sau nu.

Funcție de natura și culoarea piesei (vom putea observa mai târziu că pentru efectuarea unei mutări culoarea piesei care mută nu contează decât în cazul pionilor), precum și funcție de poziția celorlalte piese aflate pe tabla de șah, piesa poate fi mutată pe una din următoarele direcții:

- pe diagonală stânga sus,
- pe diagonală stânga jos,
- pe diagonală dreapta sus,
- pe diagonală dreapta jos,
- pe orizontală stânga,
- pe orizontală dreapta,
- pe verticală sus,
- pe verticală jos
- cele maxim opt posibilități de mutare a calului în formă de L.

Precizăm că mutările de excepție “en-passent” și rocadele regelui nu pot fi încadrate în regulile enunțate mai sus, de aceea ele vor fi tratate separat.

Fie (i, j) un câmp nou pe care se poate deplasa regulamentar piesa de pe câmpul (i_0, j_0) . Așa cum se poate vedea din figura 3, coordonatele acestui câmp pot fi deduse din relația vectorială:

$$c = c_0 + k \cdot e,$$

unde

$c = (i, j)$ este câmpul de destinație,

$c_0 = (i_0, j_0)$ este câmpul inițial,

k = este un factor de multiplicitate dat de natura piesei, iar

$e = (x, y)$ este un vector de direcție care se alege în funcție de natura piesei. Valorile posibile pentru k sunt:

$k = 1$ pentru pion, cal și rege și

$k = 1, 2, 3, \dots$ pentru damă, turn sau nebun.

		-2,-1	-2,0	-2,+1	
	-1,-2	-1,-1	-1,0	-1,+1	-1,+2
	0,-2	0,-1	i_0, j_0	0,+1	0,+2
	+1,-2	+1,-1	+1,0	+1,+1	+1,+2
		+2,-1	+2,0	+2,+1	

Fig.3 – Direcții de căutare a mutărilor posibile

Prezentăm în continuare valorile posibile ale vectorului de direcție $c = (x, y)$ și semnificația acestora (cititorul este invitat în același timp să urmărească pe figură direcțiile ale căror coordonate sunt date mai jos):

<u>Coordonate</u>	<u>Direcție</u>
(-1, +1)	diagonală dreapta sus
(-1, -1)	diagonală stânga sus
(+1, -1)	diagonală stânga jos
(+1, +1)	diagonală dreapta jos
(0, +1)	orizontală dreapta
(-1, 0)	verticală sus
(0, -1)	orizontală stânga
(+1, 0)	verticală jos
(-1, +2)	cal, dreapta-dreapta sus
(-2, +1)	cal, dreapta-sus sus
(-2, -1)	cal, stânga-sus sus
(-1, -2)	cal, stânga-stânga sus
(+1, -2)	cal, stânga-stânga jos
(+2, -1)	cal, stânga-jos jos
(+2, +1)	cal, dreapta-jos jos
(+1, +2)	cal, dreapta-dreapta jos.

Pentru introducerea acestor concepte în calculator va trebui să definim următorii vectori:

$$p = (-1, -1, +1, +1, 0, -1, 0, +1, -1, -2, -2, -1, +1, +2, +2, +1)$$

$$q = (+1, -1, -1, +1, +1, 0, -1, 0, +2, +1, -1, -2, -2, -1, +1, +2).$$

Atunci se poate observa că:

$$\begin{aligned} i &= i_0 + k \cdot p(x) \\ j &= j_0 + k \cdot q(x) \end{aligned}$$

unde x poate lua una din valorile:

- 1, 2, 3, 4, 5, 6, 7 sau 8, dacă piesa este damă sau rege, sau
- 1, 2, 3, 4, dacă piesa este nebun, sau
- 5, 6, 7, 8, dacă piesa este turn, sau
- 9, 10, 11, 12, 13, 14, 15, 16, dacă piesa este cal.

Valorile posibile pentru parametrul k au fost menționate mai sus.

Iar acum, pentru o înțelegere mai bună, vom exemplifica cele prezentate. Să presupunem că în câmpul de coordonate (i_0, j_0) se găsește un turn care trebuie să fie mutat. Rezultă că x poate lua una din valorile 5, 6, 7 sau 8, fiecare corespunzând unei direcții posibile de mutare. Dacă, de exemplu, x ia valoarea 6, găsim $p(6) = -1$ și $q(6) = 0$ deci este vorba de direcția $(-1, 0)$ care corespunde, așa cum putem vedea și din figura 3, direcției verticale sus. Pentru fiecare mutare pe această direcție se ia pe rând $k = 1, 2, 3, \dots$ până când se depășește spațiul tablei de joc sau se întâlnește o piesă de aceeași culoare sau, în sfârșit, a fost capturată o piesă de culoare adversă. Pentru $k = 1$ avem $(i, j) = (i_0 - 1, j_0)$, pentru $k = 2$ avem $(i, j) = (i_0 - 2, j_0)$, și așa mai departe, deplasând câte un câmp turnul de pe poziția (i_0, j_0) pe verticală sus.

4. Determinarea mutărilor posibile

Am văzut în lecția precedentă cum pot fi efectuate mutările pe tabla de șah. Vectorii p și q , cei care dau direcțiile de parcurs ale unei piese pe tabla de șah reprezintă de fapt acele constante determinate de natura piesei care, adăugate la coordonatele inițiale ale piesei, detectează coordonatele noului câmp pe care poate muta. Pentru a înțelege mai bine rolul pe care îl joacă acești vectori într-un program de șah, vom lua exemplul unui cal care se află pe câmpul de coordonate (4,6), adică, în notație clasică, câmpul f5. Pentru cal, variabila x poate lua una din valorile cuprinse între 9 și 16, fiecare corespunzând uneia dintre cele maxim opt mutări posibile ale acestei piese. Luând, de exemplu, $x = 9$ și ținând seamă de faptul că pentru cal k nu poate lua decât valoarea 1, determinăm astfel: $p(9) = -1$ și $q(9) = 2$, deci coordonatele noului câmp pe care poate muta calul vor fi:

$$\begin{aligned}i &= i_0 + k \cdot p(x) = 4 + 1 \cdot (-1) = 3, \\j &= j_0 + k \cdot q(x) = 6 + 1 \cdot (+2) = 8,\end{aligned}$$

adică, găsim câmpul de coordonate (3, 8), în notație clasică echivalent cu h6.

Fără îndoială, cititorul își va pune problema ce se întâmplă atunci când, prin acest procedeu matematic de determinare a coordonatelor unui câmp pe care trebuie să fie mutată o piesă, se depășește spațiul tablei de joc. Acest inconvenient poate fi eliminat foarte ușor printr-un simplu test: coordonatele (i, j) astfel determinate să fie cuprinse între 1 și 8 (de fapt, parametrii i și j nu trebuie să ia valori mai mici decât 1 sau mai mari decât 8).

Mai dificilă însă este problema determinării unei mutări posibile din punctul de vedere al regulamentului jocului de șah. Într-adevăr, trebuie avute în vedere o serie de probleme cum ar fi:

- o piesă nu poate muta pe un câmp pe care se află o altă piesă de aceeași culoare,
- cu excepția mutării calului și a rocadelor regulamentare, orice altă piesă nu poate sări peste o piesă proprie sau adversă,
- dama turnul și nebunul în mutările lor cu o deplasare mai mare de un câmp nu pot captura decât prima piesă adversă din calea lor etc.

Să le luăm pe rând.

Dama, care poate muta în toate cele opt direcții de bază (orizontale, verticale și diagonale, stânga-dreapta, sus-jos), va muta pe oricare din direcțiile alese pe orice câmp liber (câmp care conține valoarea 0) până la marginea tablei sau până la câmpul pe care se află o altă piesă, inclusiv pe acesta dacă piesa este de culoare adversă.

Regele va muta în același mod ca și dama dar numai pe câmpurile adiacente (situația cu regele aflat în șah se va trata separat).

Calul poate muta pe cele maximum opt câmpuri posibile, numai dacă acestea sunt libere sau sunt ocupate de piese de culoare adversă.

Nebunul și turnul vor muta după aceleași reguli ca ale damei, dar numai pentru patru din cele opt direcții (diagonale pentru nebun și orizontale-verticale pentru turn).

În sfârșit, pionul care, prin mutările sale posibile, poate fi încadrat parțial în câteva din categoriile expuse mai sus, cu excepția mutării “en-passent” care se tratează ca un caz special. Rocatele, de asemenea.

5. Codificarea mutărilor

Fără îndoială că partenerului de joc, dacă este un partener uman, trebuie să i se ofere posibilitatea de a furniza calculatorului mutările sale într-un mod cât mai agreabil și apropiat de cel cunoscut din practica obișnuită. Totodată, însuși computerul trebuie să-și anunțe propria sa mutare, de preferat în același mod pentru a fi înțeles de partenerul de întrecere.

În practică are loc un dialog între partenerul uman și calculator, dialog din care nu pot lipsi interpelări de genul: “Dați mutarea dumneavoastră” sau “Mutarea mea este...”. Toate informațiile cu privire la mutarea propriu-zisă sunt furnizate pe baza unui șablon care conține șase caractere, după cum urmează:

$L_1C_1-L_2C_2X$,

unde L_1 și L_2 reprezintă o literă de la a la h, C_1 și C_2 reprezintă o cifră de la 1 la 8. Cu alte cuvinte, L_1C_1 și L_2C_2 reprezintă notațiile clasice ale câmpului din care pleacă piesa care mută, respectiv ale câmpului în care ajunge această piesă.

Liniuța despărțitoare de pe poziția a treia poate fi înlocuită cu un spațiu sau, atunci când este vorba de o bătaie de piesă adversă, ea poate fi înlocuită cu simbolul cunoscut și utilizat în notația clasică “:”.

Ultimul caracter, **X** de pe poziția a șasea, de regulă nu se completează. El va fi utilizat numai în cazuri speciale, atunci când, de exemplu, s-a făcut o transformare de pion, și va trebui să se precizeze în ce piesă anume se transformă respectivul pion, deci în poziția a șasea se va completa, după caz, una din literele: **D** pentru damă, **T** pentru turn, **N** pentru nebun sau **C** pentru cal. De asemenea **X** mai poate fi completat cu litera **P** (pion). În acest caz se are în vedere că este vorba despre o mutare de pion cu luare “en-passent” posibilă, conform mutării precedente a adversarului. Un ultim caz de utilizare a celui de-al șaselea caracter din codificarea mutării este cel în care se folosește litera **R** (rege), semnificând că este vorba de efectuarea unei rocade. De data aceasta cele două câmpuri (L_1C_1 inițial și L_2C_2 final) vor constitui câmpurile de pe care mută regele și, respectiv, în care ajunge regele după efectuarea regulamentară a rocadei. Tipul rocadei (mare sau mică) este determinat automat, în funcție de destinație. Iată în continuare câteva exemple de mutări:

e2-e4 : piesa de pe câmpul e2 mută pe câmpul e4;

e7:e8D : pionul de la e7 bate piesa aflată pe câmpul e8 și se transformă în damă;

e8-e8R, e8-g8R : rocada mare, respectiv rocada mică a regelui negru;

g7-g5P : mutare de pion de pe câmpul inițial g7 pe câmpul g5, cu luare “en-passent” a pionului de pe câmpul f6 sau h6, funcție de ultima mutare a adversarului.

Este de observat că nu este necesar să se precizeze natura piesei care mută, aceasta determinându-se automat pe baza adresei furnizate (L_1C_1). Se înțelege că acceptarea unei mutări de către computer trebuie să fie precedată de o validare riguroasă a respectivei mutări. În caz contrar, poziția de pe tabla de șah poate să fie compromisă, iar partida, mai devreme sau mai târziu, va fi anulată.

Trebuie înțeles de asemenea că, după asimilarea unei mutări, computerul va trebui să-și transforme cu grijă toate elementele recepționate în coduri numerice proprii și să rețină toate informațiile. Validarea mutărilor primite de un adversar se face în spiritul regulamentului. Orice abatere de la o regulă cunoscută în jocul de șah va fi returnată adversarului cu un mesaj de avertizare și i se va cere acestuia să reintroducă mutarea sa, până când computerul constată că a recepționat o mutare corectă și poate să treacă la propriile sale analize pentru a trimite un răspuns.

6. Generatorul de mutări

Înainte de a trece la subiectul anunțat în titlul acestei lecții considerăm că este necesar să spunem câteva cuvinte despre rolul pe care îl deține în controlarea jocului de către computer un element-pivot care este notat, de exemplu, cu litera **O** și care poate lua valorile +1 pentru cazul în care calculatorul joacă cu piesele albe, și -1 pentru cazul contrar. Acest element intervine aproape în toate calculele și transformările necesare atunci când trebuie decisă apartenența unei piese sau a unor categorii de mutări. În acest sens prezentăm un exemplu care, deși nu este cel mai semnificativ, poate fi ușor de înțeles. Produsul **O-T(i, j)** va fi totdeauna pozitiv dacă piesa de pe câmpul de coordonate (i, j) aparține calculatorului, și este negativ dacă piesa respectivă aparține adversarului.

Am văzut în lecțiile anterioare cum poate fi executată o mutare legală. Să presupunem că avem memorată în calculator tabla de șah conținând o poziție oarecare a pieselor, la poziția la care s-a ajuns după un schimb de mutări între calculator și partenerul său de joc, sau care a fost introdusă inițial pentru a fi continuată. Cum ar trebui să procedeze calculatorul pentru a efectua o mutare bună?

Abia acum apare ca fiind necesară o analogie între posibilitățile de “a gândi” ale calculatorului și felul în care gândește și acționează în această situație un jucător uman.

Un jucător uman, așa cum, de altfel, recomandă majoritatea cărților bune de teorie șahistă, va începe prin a analiza poziția de pe tabla de șah. În această situație el va selecta, din mulțimea mutărilor posibile, pe acelea care i se par cele mai adecvate configurației de pe tabla de șah. În practică s-a demonstrat că acestea nu sunt, de regulă, mai mult de 10. De asemenea, dintre acestea mutări candidat, cele mai importante (printre care ar trebui să se afle, probabil, și cea care va fi aleasă de jucător), nu sunt mai mult de două-trei. În funcție de timpul de gândire de care dispune și de forța de analiză a jucătorului respectiv, acesta va analiza de aici înainte, pentru fiecare din mutările sale candidat, răspunsurile posibile ale adversarului. Practic, el va gândi în mod asemănător și pentru adversar, primele răspunsuri posibile, și așa mai departe, până când, la capătul unor variante de mutări (de fapt, semimutări) alternativ generate, va putea să decidă dacă stă mai bine sau mai rău, sau s-a ajuns la o situație de egalitate în poziția rezultată. În final el va hotărî asupra mutării pe care urmează să o execute și, în cazul în care nu mai dispune de timp suficient de verificare a calculelor, sau dacă este convins că mutarea aleasă este cea mai bună, o va executa.

În principiu, computerul ar trebui să urmeze același raționament pentru a rezolva problema alegerii mutării optime. În realitate există câteva excepții deloc lipsite de semnificație. De exemplu, jucătorul uman, dintr-o singură privire, elimină din calcul aproape toate mutările care sunt imposibile sau par absurde și, într-o proporție covârșitoare, aceste eliminări se dovedesc a fi corecte. Văzând un pion blocat sau o

piesă care, dacă ar fi mutată de pe poziția pe care o acupă ar duce la dezastru sau la situații sensibil nefavorabile, jucătorul uman decide în câteva fracțiuni de secundă că nu trebuie să mai continue analiza sa pe seama acestor variante. În schimb, calculatorul trebuie să decidă în astfel de situații analizând până la capăt imposibilitatea efectuării unei mutări sau căutând unele implicații care pot surveni până acolo când poate lua o hotărâre certă. El nu “vede” tabla de șah decât controlând-o în amănunțime și calculând toate implicațiile conform algoritmului implementat.

În consecință, calculatorul trebuie să țină seama de tot ceea ce se petrece pe tabla de șah, fără să poată elimina nici măcar o mutare doar “privind-o”, ci numai după o analiză laborioasă supusă unor calcule minuțioase, deloc simple.

Să luăm ca exemplu acel pion blocat despre care spuneam că jucătorul uman aproape că nici nu-l “observă” în analiza sa. Calculatorul decide asupra imposibilității mutării acestuia abia după ce testează toate câmpurile pe care prezunmtiv ar putea fi mutat, inclusiv cele legate de mutări de tip “en-passent”.

7. Generatorul de mutări. Nivelul de joc

Generarea mutărilor posibile se face, de regulă, printr-o subrutină specializată care, după ce validează în prealabil posibilitatea execuției lor, generează mutările posibile după o parcurgere, secvențială sau nu, a tablei de șah.

Generarea se face alternativ pentru alb și negru, creându-se astfel variante de mutări cu o lungime stabilită anterior sau decisă de criterii prestabilite.

Să presupunem că la prima semimutare sunt generate n_1 mutări. Să mai presupunem în continuare că adversarul, pentru fiecare din cele n_1 mutări, poate răspunde cu n_2 mutări și așa mai departe, până la semimutarea cu numărul k în care, pentru fiecare din cele n_{k-1} mutări ale adversarului se poate răspunde cu n_k mutări. În total vor fi elaborate în cursul procesului de generare un număr de $N = n_1 \cdot n_2 \cdot n_3 \dots n_k$ variante de lungime k și vor fi executate un număr de $M = n_1 + n_1 \cdot n_2 + n_1 \cdot n_2 \cdot n_3 + \dots$ mutări. Avem în acest mod de-a face cu un arbore de mutări cu M noduri și N ramuri de lungime maximă.

Dacă generarea mutărilor s-ar face exhaustiv, atunci variantele generate ar fi suficient de lungi pentru a putea întâlni la capătul lor situații concrete bazate pe criterii de finalitate ale jocului (mat, pat, remiză). Dacă toate operațiunile necesare acestor generări de mutări și analizelor pozițiilor rezultate ar putea fi acoperite într-un timp rezonabil de răspuns al calculatorului, atunci un astfel de program ar rezolva definitiv problema jocului de șah și, fără îndoială, ar câștiga aproape în fața oricărui jucător uman. Însă lucrurile nu stau nici pe departe așa. Și vom vedea imediat care sunt adevăratele motive pentru care nu s-a putut rezolva până în prezent problema jocului de șah pe calculator la modul definitiv.

În practică s-a dovedit că numărul mediu de mutări posibile dintr-o poziție oarecare de pe tabla de șah este de aproximativ $n = 40$. Dacă am considera variantele generate doar pentru $m = 10$ semimutări (cinci perechi de mutări alternative alb-negru), am avea de-a face cu un număr de $N = n^m = 40^{10}$ variante, adică un număr extraordinar de mare (conține nu mai puțin de 17 cifre) pe care nici un calculator din lume la ora actuală nu l-ar putea "acoperi" în timp util, ținând seamă și de faptul că o variantă, în mod practic, presupune ea singură un număr foarte ridicat de operații elementare bazate pe calcule numerice, comparări sau alegeri.

Pornind de la imposibilitatea tratării exhaustive a problemei jocului de șah pe computer și de la alte probleme asemănătoare s-a ajuns, după lungi căutări și frământări ale promotorilor, la elaborarea unei discipline științifice noi intitulată inteligența artificială care și-a propus în principal transformarea computerului, dintr-o mașină brută, capabilă doar să calculeze ore în șir după "scheme" prestabilite, într-o unealtă capabilă să dezvolte raționamente asemănătoare cu cele emise de mințile omenești.

Despre numărul $N = n^m$ se discută mult astăzi în literatura de specialitate. În programele de șah n și m se aleg destul de mici pentru a se putea profita de un timp

de răspuns rezonabil al calculatorului. Numărul n mai este numit și orizontul analizei, iar m mai este cunoscut și sub denumirea de adâncime a analizei.

În general, un program de șah poate lua în seamă mai multe valori ale perechii (n,m) . Acest fapt caracterizează de regulă nivelul de joc al programului care se materializează până la urmă în calitatea jocului mașinii. Acest nivel se fixează înainte de începerea jocului, dar sunt și cazuri când el poate fi modificat în timpul jocului chiar de către program, funcție de anumite elemente sau situații care intervin pe parcursul desfășurării partidei. Se înțelege că un nivel de joc este cu atât mai ridicat (practic, calitatea jocului computerului este cu atât mai bună), cu cât orizontul și/sau adâncimea analizei sunt mai mari.

8. Funcția de evaluare

Să presupunem că avem o poziție dată P_0 și că, în urma unei alegeri (după anumite criterii al căror conținut nu ne interesează deocamdată), s-a efectuat o mutare a unei piese de pe câmpul $T(i_0, j_0)$ pe câmpul $T(i, j)$. A rezultat o nouă poziție P_1 în care au intervenit situații noi. Prin ce diferă cele două poziții? Are oare importanță faptul că piesa de pe câmpul inițial a fost pusă pe câmpul final și de ce? Ce caracterizează în fond o mutare pe tabla de șah?

Acestea sunt doar câteva întrebări la care răspunsul nu este deloc simplu de dat, dar care se pun ori de câte ori suntem interesați să introducem într-un program de șah anumite criterii care să evalueze și să compare mutările. Un singur lucru este foarte clar: în urma unei mutări efectuate se schimbă unele "relații" pe tabla de șah, care de fapt reflectă interdependența pieselor și controlul câmpurilor pe tabla de șah și alcătuiesc, prin diferența dintre cele două poziții P_1 și P_0 un cumul de avantaje sau dezavantaje care caracterizează oportunitatea sau inoportunitatea mutării.

Să presupunem acum că dispunem de un sistem de apreciere numerică pentru "importanța" unor astfel de relații. Acest sistem poate fi conceput sub forma unor tablouri matriceale de punctaje în modul următor:

- tabloul relațiilor de atac, $at(x, y)$, în care sunt păstrate valorile numerice atribuite unor relații de tipul "*piesa x atacă piesa adversă y*",
- tabloul relațiilor de apărare, $ap(x, y)$, în care sunt păstrate valorile numerice atribuite unor relații de tipul "*piesa x apără piesa y*" (se înțelege că ambele piese sunt, în acest caz, de aceeași culoare), și
- tabloul relațiilor de control a câmpurilor libere, care conține valorile numerice ale relațiilor de tipul "*piesa x controlează un câmp liber*".

Disponând de toate aceste date se înțelege că orice mutare efectuată pe tabla de șah poate fi "apreciată" numeric prin diferența cumulativă a tuturor relațiilor noi intervenite în poziția P_1 și a celor "pierdute" din poziția anterioară P_0 în urma efectuării mutării.

Pentru a înțelege mai bine, să studiem anumite exemple. Considerăm poziția P_0 ca fiind poziția inițială a jocului de șah, iar poziția P_1 ca fiind cea care survine după prima mutare a albului, și anume e2-e4.

În poziția inițială pot fi semnalate o mulțime de relații de genul acelor anunțate mai înainte, dintre care amintim câteva:

- calul alb din b1 apără pionul din d2; același cal controlează două câmpuri libere, cele din c3 și a3.
- turnul negru din h8 apără calul din g8 și pionul din h7 etc.

Să enumerăm relațiile importante legate în exclusivitate de mutarea anunțată e2-e4 care transformă poziția P_0 în poziția P_1 . Față de mulțimea relațiilor existente în poziția P_0 , odată cu efectuarea mutării e2-e4 intervin următoarele modificări:

- dispar relațiile:

- pionul e2 controlează câmpurile libere d3 și f3
- calul din g1 apără pionul din e2
- nebunul din f1 apără pionul din e2
- regele din e1 apără pionul din e2
- dama din d1 apără pionul din e2,

- apar relațiile:

- calul din g1 controlează câmpul liber din e2
- nebunul din f1 controlează câmpurile libere din e2, d3, c4, b5 și a6
- regele din e1 controlează câmpul liber din e2
- dama din d1 controlează câmpurile libere din e2, f3, g4 și h5
- pionul din e4 controlează câmpurile libere din d5 și f5.

Toate celelalte relații existente pe tabla de șah în poziția inițială P_0 rămân neschimbate și după efectuarea mutării e2-e4, adică în noua poziție P_1 .

Prin diferență (totalitatea punctelor obținute prin apariția noilor relații din care se scad cele pierdute prin dispariția relațiilor menționate) rezultă așa numita “notă” a mutării care, prin comparație cu altele, determină până la urmă oportunitatea sau inoportunitatea mutării studiate. Evident, vor trebui luate în seamă doar acele mutări care sunt cotate cu un punctaj maxim.

Mai trebuie menționat și faptul că relațiile despre care s-a vorbit mai sus pot fi favorabile calculatorului sau adversarului. Astfel că, dacă o anumită mutare conduce la apariția unor relații favorabile adversarului, punctajul trebuie diminuat în mod corespunzător, iar dacă sunt distruse anumite relații favorabile adversarului, atunci punctajul trebuie să fie mărit corespunzător cu valoarea absolută a acestora. Astfel spus, determinarea “notei” unei mutări se face printr-o însumare algebrică (cu *plus* pentru relațiile proprii noi intervenite și pentru dispariția unor relații ale părții adverse, și cu *minus* pentru dispariția unor relații proprii și pentru apariția unor relații noi favorabile adversarului).

După cum se poate înțelege, tablourile de punctaje despre care s-a amintit mai sus trebuie în mod necesar să conțină o apreciere cât mai justă a potențialului fiecărei relații posibile pe tabla de șah. Astfel, relația de atac de exemplu în cazul unei dame, trebuie să fie mai bine cotată decât cea corespunzătoare unui atac a aceleași figuri asupra unui pion sau, controlul unei piese asupra unui câmp central, în special în fazele inițiale ale partidei, trebuie să fie mai bine punctat decât cel asupra unui câmp necentral etc.

Cele prezentate mai sus reprezintă o concepție originală pentru ceea ce se cheamă “funcție de evaluare” în șahul-computer. Punerea ei în practică, adică implementarea acestei idei într-un program de șah conduce, așa cum ușor se poate presupune, la un joc al calculatorului cu “tendința” de dominare, de sufocare a adversarului, prin acumularea a cât mai mult spațiu de joc, prin constrângerea pieselor adverse, prin cooperarea și mobilitatea figurilor proprii etc., mai ales în anumite faze ale jocului.

Marea dificultate în utilizarea unei astfel de soluții o reprezintă stabilirea riguroasă a notelor atribuite fiecărei relații potențiale care poate interveni pe tabla de șah. Există mai multe căi de determinare a acestor punctaje. Una dintre ele este cea de natură statistică sau apelând la metode stochastice. Există cu siguranță în unele poziții de pe tabla de șah anumite mutări care pecetluiesc sau compromit definitiv soarta partidei. Acestea sunt mutările care trebuie să stea în atenția noastră atunci când ne propunem să determinăm cât mai realist notele care trebuie acordate unor relații de atac, apărare sau control. Evaluând prin diferență asemenea mutări în poziții distincte, în comparație cu altele obișnuite, se poate ajunge la un sistem de punctaj care să dea satisfacție. Calea este destul de dificilă, dar trebuie reținut că,

până în prezent, ea nu a fost explorată suficient.

O altă metodă de determinare a unei funcții de evaluare, care face parte din arsenalul celor mai moderne dar foarte rar și dificil de abordat, constă în folosirea unei tehnici de autoinstruire a programului, care "îmbunătățește" sistemul de evaluare pe măsură ce acesta este supus experimentărilor. În mod practic se pornește de la un sistem de evaluare în care notele au fost alese arbitrar și, pentru fiecare partidă nouă și fiecare mutare care nu conduce la o îmbunătățire a situației, se modifică punctajul în mod corespunzător. Se zice, în astfel de cazuri, că programul "învață" din propria-i experiență.

Funcția de evaluarea în șahul-computer reprezintă de fapt cheia descifrării tuturor enigmelor care mai persistă și astăzi în problema jocului de șah. Majoritatea programelor care tratează această problemă au la bază implementarea unor tehnici și strategii de joc cunoscute din practica șahistă:

- controlul centrului,
- mobilitatea figurilor,
- ocuparea coloanelor deschise sau semideschise,
- avantajul perechii de nebuni,
- atacul la f7 sau f2, h7 sau h2,
- jocul pionului izolat,
- regula pătratului,
- cooperarea figurilor etc.

Toate acestea, oricât de multe dintre ele ar fi implementate și respectate într-un program de șah, nu pot reuși să justifice complet tot ceea ce caracterizează oportunitatea unei mutări în șah.

9. Alegerea mutării optime

Așa cum s-a putut înțelege până acum, alegerea celei mai bune mutări reprezintă factorul determinant care caracterizează diversitatea algoritmilor implementați pe calculator. Jocul unui program este caracterizat în principal de “calitatea” funcției de evaluare pe care o utilizează acesta în alegerea mutării optime.

În elaborarea variantelor de analiză pentru căutarea celei mai bune mutări se generează, așa cum am și aflat deja, un arbore de joc al poziției, format din noduri care conțin mutări alternative ale albului și negrului și ramuri care reprezintă variante posibile de joc. Un astfel de arbore, care va constitui pentru noi un exemplu pentru cele ce urmează, este cel reprezentat în figura 4 și care corespunde următoarei poziții simplificate de pe tabla de șah: alb: Rb6, a5, b3, c5, c6 și negru: Rb8, a6, b7.

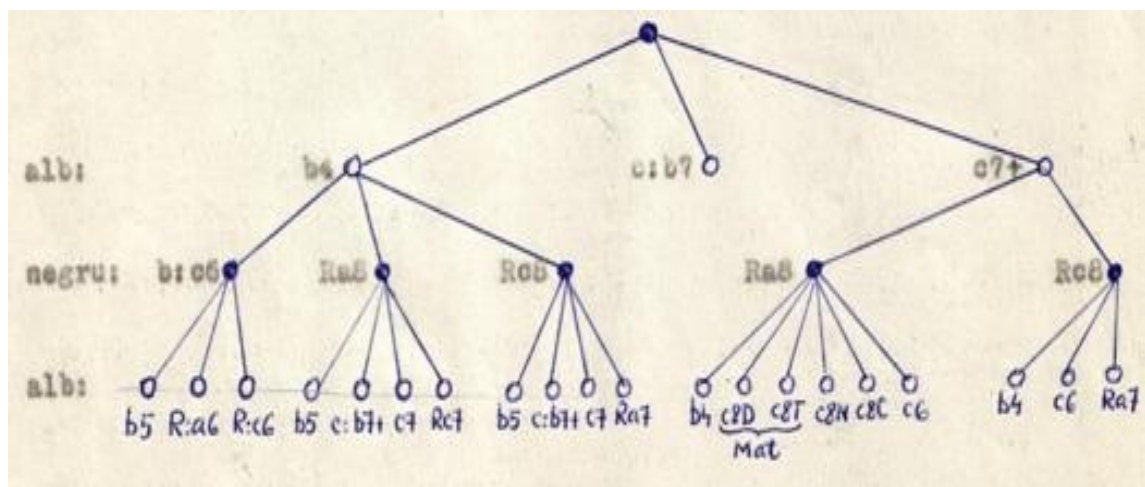


Fig.4. Arborele de joc al poziției

Pot fi observate atât mutările alternative alb-negru care corespund nodurilor arborelui, precum și diferitele ramuri formate de acestea, care constituie variantele posibile de joc.

O desfășurare completă (până la capăt) a acestui arbore de joc pentru poziția dată poate conduce la găsirea unui răspuns potrivit pentru stingerea celui mai important obiectiv urmărit într-o partidă de șah – matul – dar, așa cum am mai subliniat într-una din lecțiile anterioare, este imposibil în mod practic să realizezi această performanță, mai ales în pozițiile care conțin modalități largi de mutare ale pieselor.

După cum se poate ușor intui, în practica jocului se manifestă două tendințe opuse: în timp ce, de exemplu, albul urmărește să găsească cea mai bună

mutare a sa, negrul va căuta pentru el tot cea mai bună mutare care, de fapt, este cea care diminuează cel mai mult tendința adversarului. În acest caz se impune cu necesitate intervenția unui compromis și anume, albul va alege, din multitudinea de mutări pe care le are la dispoziție, pe aceea pe care negrul o poate contracara cel mai puțin. Așadar, în timp ce albul, care este la mutare, caută un “maxim” calitativ sau cantitativ prin mutarea sa, interesul adversarului, asemănător cu al său, îl determină să aleagă doar acea mutare care poate fi combătută de adversar cu un “minim” de efect.

Algoritmul adecvat în asemenea situații, pentru rezolvarea acestor categorii de probleme, este cunoscut sub denumirea de “**minimax**”. Pentru simplificarea aplicării acestuia în jocul de șah este necesar ca evaluările pentru mutările calculatorului să se facă cu numere pozitive iar cele ale adversarului cu numere negative. Deoarece algoritmul “minimax” poate fi întâlnit astăzi în multe materiale aparținând literaturii de specialitate nu-l vom prezenta aici în detaliu. În orice caz, ideea de bază a acestuia a fost sugerată mai sus, iar analiza completă a exemplului oferit îl poate sprijini pe cititorul interesat pentru constituirea pașilor care compun un asemenea algoritm.

Aplicarea exhaustivă a algoritmului “minimax” se dovedește deseori insuficientă, mai ales atunci când orizonturile și adâncimile analizei variantelor generate sunt mari. De aceea, algoritmul este însoțit aproape întotdeauna de diferite criterii de reducere a numărului de ramuri de parcurs ale arborelui generat pentru analiză. Cel mai important criteriu de acest gen se numește “*principiul alfa-beta*” și are la bază ideea că, în parcurgerea secvențială a variantelor generate, este foarte important ca mutările cele mai bune (cele cu note mari) să fie analizate primele. Acest fapt presupune o ordonare a mulțimii de noduri de pe același nivel, întrucât o variantă care conține noduri bine notate are șanse mai mari de a duce spre o situație favorabilă. Practic, multe mutări pot aduce un dezavantaj mare și ele nu pot fi ocolite decât dacă pot fi comparate cu altele mai bune, care deja au trecut prin ciurul analizei. Un astfel de principiu introdus în algoritmul “minimax” conduce la o reducere substanțială a numărului de variante calculate din arborele de joc.

Un alt criteriu utilizat în reducerea numărului de operații în generarea arborelui de joc al poziției este denumit *criteriul “razoring”* și se bazează pe ideea că în jocul de șah nu se poate spera ca prin mutarea adversarului să fie îmbunătățită poziția proprie. Criteriul este riscant, întrucât este bine știut că nu totdeauna adversarul răspunde cu mutarea cea mai bună. Însă, pentru scopul anunțat, acela de a câștiga timp de gândire prin eliminarea anumitor calcule inutile, criteriul se poate dovedi foarte util.

În fine, un alt criteriu folosit are la bază ideea că o mutare proprie nu este oportună dacă nu aduce o îmbunătățire a propriei poziții. Acesta se numește *criteriul “better”* și, așa cum ușor s-ar putea înțelege, nici el nu este prea potrivit pentru desfășurarea neîngrădită a unei partide de șah.

10. Tratarea excepțiilor în șahul programat

După cum s-a mai menționat, nu toate regulile jocului de șah pot fi tratate uniform. Cel puțin în privința anumitor mutări lucrurile pot părea destul de dificile și pot ridica destule probleme. De aceea considerăm necesar că ar fi binevenite câteva sugestii și orientări în această direcție.

Pentru includerea mutărilor de excepție într-un program de șah este necesar să definim anumiți parametri, care vor juca un rol specific important pe tot parcursul desfășurării partidei. În continuare vom încerca pe rând toate cazurile care fac obiectul acelor probleme și vom prezenta unele modalități de soluționare a lor.

a) Rocatele

Pentru rocada mică se definesc următorii parametri:

- R_1^+ , pentru alb și R_1^- pentru negru, care conțin valoarea 0 dacă rocada mică nu s-a efectuat sau se poate efectua, și valoarea 1 dacă rocada mică s-a efectuat sau nu se mai poate efectua.

În mod analog se definesc și parametrii pentru rocada mare:

- R_2^+ pentru alb și R_2^- pentru negru.

La începutul sau pe parcursul desfășurării jocului se va ține seama de următoarele observații:

- în poziția inițială cei patru parametri trebuiesc inițializați cu valoarea 0.
- dacă partida începe de la o anumită poziție, alta decât cea inițială, parametrii rocadelor trebuie să fie inițializați cu valorile corespunzătoare (0 dacă se mai poate efectua rocada, respectiv 1 în caz contrar).
- în timpul desfășurării partidei, dacă turnul damei (pentru rocada mare) sau turnul regelui (pentru rocada mică) a fost mutat, imediat se atribuie valoarea 1 parametrului corespunzător R_2^+ sau R_2^- , respectiv R_1^+ sau R_1^- .
- în timpul desfășurării partidei, dacă s-a mutat regele, se actualizează cu valoarea 1 parametrii corespunzători ambelor rocade R_1^+ și R_2^+ pentru alb, sau R_1^- și R_2^- pentru negru.
- Pentru efectuarea rocadei mici este necesar să se verifice în prealabil dacă:
 - parametrul corespunzător rocadei respective are valoarea 0 (se poate efectua rocada),
 - câmpurile T(8, 6) și T(8, 7) pentru alb și, respectiv T(1, 6) și T(1, 7) pentru negru sunt libere (au valoarea 0) și nu sunt controlate de o piesă adversă.
- Efectuarea rocadei mici presupune următoarele operațiuni:
 - pentru alb: $T(8, 7) = +1$, $T(8, 6) = +3$, $T(8, 5) = T(8, 8) = 0$, $R_1^+ = 1$
 - pentru negru: $T(1, 7) = -1$, $T(1, 6) = -3$, $T(1, 5) = T(1, 8) = 0$, $R_1^- = 1$.

- Pentru efectuarea rocadei mari este necesar să se verifice în prealabil dacă:
 - parametrul corespunzător rocadei respective are valoarea 0 (se poate efectua rocada),
 - câmpurile $T(8, 2)$, $T(8, 3)$ și $T(8, 4)$ pentru alb și, respectiv $T(1, 2)$, $T(1, 3)$ și $T(1, 4)$ pentru negru sunt libere (au valoarea 0) și nu sunt controlate de o piesă adversă.
- Efectuarea rocadei mari presupune următoarele operațiuni:
 - pentru alb: $T(8, 3) = +1$, $T(8, 4) = +3$, $T(8, 1) = T(8, 5) = 0$, $R_2^+ = 1$,
 - pentru negru: $T(1, 3) = -1$, $T(1, 4) = -3$, $T(1, 1) = T(1, 5) = 0$, $R_2^- = 1$.

b) Mutări de pioni

b1) Mutări de înaintare:

- se face $j = j_0$ și $l = i_0 - 1$ pentru alb, sau $l = i_0 + 1$ pentru negru,
- dacă câmpul $T(i, j)$ nu conține valoarea 0 se trece la b2).
- se consideră mutare posibilă și se efectuează operațiunile: $T(i, j) = T(i_0, j_0)$ și $T(i_0, j_0) = 0$,
- se verifică dacă mutarea este necesară, altfel urmează căutarea unei alte mutări după cum urmează:
- se verifică dacă i_0 este linia inițială de mutare a pionului, caz în care pionul poate executa un salt de două câmpuri pe coloană: deci, dacă i_0 nu este 7 pentru alb, respectiv i_0 nu este 2 pentru negru, se trece la pasul b2),
- se face $j = j_0$ și $l = i_0 - 2$ pentru alb sau $l = i_0 + 2$ pentru negru,
- dacă $T(i, j)$ nu conține valoarea 0 se trece la pasul b2),
- în caz contrar se consideră mutare posibilă și se efectuează următoarele operațiuni: $T(i, j) = T(i_0, j_0)$ și $T(i_0, j_0) = 0$.

b2) Mutări de capturare:

- se face $l = i_0 - 1$ pentru alb sau $l = i_0 + 1$ pentru negru, iar $j = j_0 \pm 1$ (pe rând),
- dacă $T(i, j)$ nu conține o piesă de culoare adversă se trece la pasul b3),
- în caz contrar se consideră mutare posibilă de capturare și se efectuează operațiunile: $T(i, j) = T(i_0, j_0)$ și $T(i_0, j_0) = 0$.

b3) Mutări "en-passant":

- Se bazează pe semimutarea anterioară, deci pe mutarea anterioară a adversarului; dacă această mutare a fost una de pion de pe câmpul său inițial cu avansare imediată de două pătrățele pe coloană și dacă a trecut prin bătaia pionului propriu și dacă (i_0, j_0) sunt coordonatele câmpului pionului propriu, atunci se face : $T(i_0, j_0 \pm 1) = 0$ (se capturează pionul advers, după caz), apoi $j = j_0 \pm 1$, $l = i_0 - 1$ pentru alb, sau $l = i_0 + 1$ pentru negru, $T(i, j) = T(i_0, j_0)$ și $T(i_0, j_0) = 0$.

Pentru a ști că mutarea anterioară a adversarului a fost o mutare inițială de pion cu salt de două câmpuri pe coloană (caz în care ar putea fi posibilă o mutare "en-passant") se va folosi câte un indicator special, pentru alb și pentru negru, respectând pentru inițializare și punere la zi a acestora reguli analoage cu cele de la rocade.

b4) Transformări de pioni:

Dacă într-una din situațiile b1) sau b2) pionul ajunge pe ultima sa linie de înaintare (linia 1 pentru alb și, respectiv linia 8 pentru negru), acesta poate fi transformat în damă D (damă), T (turn), N (nebun) sau C (cal), pe rând, fiecare din acestea considerându-se o mutare. Pentru aceste cazuri operațiunile care trebuiesc executate constă în:

$$T(i, j) = +/-1, +/-2, +/-3, \text{ sau } +/-4 \text{ și } T(i_0, j_0) = 0.$$

Evident, pot fi găsite căi mult mai simple de efectuare a acestor mutări de excepție. În cele prezentate mai sus s-a ales calea care, din punct de vedere intuitiv, poate părea mult mai ușor de înțeles.

11. Fișiere de deschidere și finaluri

De regulă, programele de șah pe calculator sunt concepute pentru a răspunde celor mai multe dintre cerințele competiționale. Una dintre aceste cerințe este cea legată de timpul de răspuns. În general, pentru efectuarea unei mutări se dispune de circa două-trei minute. Evident, acest timp nu este împărțit uniform pe întreaga durată a partidei sau în toate fazele ei. Din practica șahiștilor profesioniști se știe că în faza de deschidere mutările se fac mai rapid, jucătorii venind, de regulă, cu “lecția învățată de acasă”. Prin aceasta se înțelege de fapt că șahiștii profesioniști stăpânesc o bună parte dintre deschideri sau variante ale acestora, bine puse la punct, astfel că, în fața tablei de șah ei pot alege o cale sau alta, funcție de preferințele proprii sau ale adversarului etc. Efectul acestor “memorări” se manifestă și prin aceea că, primele mutări derulându-se rapid, se beneficiază de o rezervă importantă de timp care poate fi de mare folos în fazele mai complicate ale jocului.

Așa cum era de așteptat, creatorii de programe de șah au speculat chiar de la început această idee. Ei au conceput, în diverse variante, ceea ce se cheamă astăzi în literatura de specialitate, **fișierul de deschideri**. Acesta constă dintr-un arbore de joc care conține principalele mutări și variante din faza de deschidere. Toate mutările și variantele sunt memorate astfel încât, atunci când adversarul a făcut o mutare care corespunde unei variante concrete din acest fișier, calculatorul, utilizând un sistem facil de căutare bazat pe legături (înlănțuiri), va răspunde imediat cu mutarea următoare dacă va fi găsită. Aceste căutări se petrec până când una din mutările adversarului “sare” din arborele variantelor memorate, sau până când s-a ajuns la capătul acestora. Se înțelege că variantele memorate pot fi destul de lungi, dar în practică un fișier de deschideri poate cuprinde abia câteva zeci de mii de mutări, ceea ce face ca utilitatea lui să se manifeste în majoritatea cazurilor doar la primele trei-patru mutări ale jocului. Pe de altă parte, ideea utilizării unui fișier de deschidere cât mai complet nu poate fi și nici n-ar putea vreodată fi o rezolvare “onorabilă” a problemei șahului-computer. El ajută doar, așa cum s-a menționat, la câștigarea unui timp de gândire, care poate fi foarte prețios (din punct de vedere competițional), întrucât poate deveni foarte necesar în anumite momente mai dificile ale partidei, pentru o analiză mai detaliată a unor situații complicate intervenite pe tabla de șah.

În aceeași idee sunt folosite uneori și fișiere de finaluri, care conțin rezolvări, prin același mecanism, ale unor finaluri cunoscute din teoria și practica jocului de șah. De exemplu: finalurile de turn și rege contra rege, cal, nebun și rege contra rege, doi nebuni și rege contra rege, cele de pioni etc., sunt doar câteva dintre cele care pot fi incluse într-un fișier de finaluri.

12. Considerații finale

Nu există creatori de programe de șah care să nu-și pună problema evaluării numerice a pieselor din componența acestui joc. Maestrul britanic Howard Staunton a evaluat valoarea acestora, printr-un procedeu contestat astăzi, luând ca unitate de apreciere valoarea unui pion. El a oferit următoarele evaluări: un turn este egal cu 5,48 pioni, un cal este egal cu 3,05 pioni, un nebun este egal cu 3,50 pioni și o damă este echivalentă cu valoarea a 9,94 pioni. Există însă multe alte evaluări care privesc aspectul material al obiectivelor urmărite în jocul de șah, dar toate pot fi combătute ușor. Marea dificultate constă de fapt în evaluarea unei piese în conjunctură cu poziția ei și cu celelalte piese de pe tabla de șah, ceea ce sugerează ideea unei evaluări dinamice. Subiectul este destul de larg și nu ne propunem să-l dezbatem complet aici. Cert este faptul că, fără a lua în seamă evaluarea atunci când îți propui să abordezi rezolvarea problemei jocului de șah pe calculator, este practic imposibil să obții vreun succes. Subiectul deschide în mod necesar tema existenței unei funcții de evaluare cât mai eficiente, de fapt, marea necunoscută a șahului-computer.

Ideea evaluării pe baza atribuirii unor note statice, bonificații sau penalizări pentru piese, strategii, avantaje, dezavantaje, câmpuri etc. este din start sortită eșecului. Chiar dacă și în acest mod se poate ajunge la un program de șah care să atingă anumite performanțe și chiar să câștige în fața unui mare-maestru, nu acesta va conduce la rezolvarea deplină a problemei. Pentru că, ar trebui să se înțeleagă foarte bine, nu înfrângerea unui mare maestru, fie el și campion mondial, este obiectivul cel mai important pe care și l-a propus omul atunci când s-a gândit la definitivarea problemei șahului electronic. De la bun început omul a sperat că, învingând problema șahului computer, va putea să învingă apoi numeroase altele care îi stau în cale de ani și ani.

De sute de ani o mulțime de minți luminate nu au reușit să se decidă asupra unui sistem de puncte pentru cele șase piese care fac obiectul jocului de șah. Imaginați-vă că mâine, un asemenea sistem va fi anunțat cu mare pompă că a fost descoperit și că un computer învinge aproape de fiecare dată campionul mondial al jocului de șah. La ce ar mai fi bun acel sistem de puncte care ajută cu exactitate la evaluarea unei mutări? Cu puterea sa de analogie și capacitatea sa creativă, omul îl va implementa în diferite alte domenii, unde i-ar putea aduce suficientă satisfacție. Dar tot omul este acela care nu se va mulțumi niciodată cu jumătățile de măsură. El va dori în continuare perfecțiunea și va fi dezamăgit ori de câte ori sistemul nu se va comporta pe măsura așteptărilor. Iată un scurt exemplu de ingraturitate manifestată de om în fața jumătăților de măsură, luat din domeniul matematicii, practic tot din domeniul rezolvării problemelor. De trei sute de ani aproape toți marii matematicieni ai lumii și-au bătut capul cu rezolvarea unei probleme cunoscută sub numele de "marea teoremă a lui Fermat". Au fost instituite și numeroase premii, s-au consumat milioane de coli de hârtie cu demonstrații care n-au satisfăcut. Și iată că, în urmă cu câțiva ani s-a anunțat rezolvarea "aproape completă" a problemei de către un tânăr

matematician german numit Fastings. Era un mare pas înainte, unii au îndrăznit să afirme, nu fără oarecare umor, că problema a fost rezolvată doar în proporție de 99,99%. Și totuși, nici un matematician serios nu consideră astăzi că problema a fost rezolvată definitiv. Ea mai stă încă în atenția specialiștilor și mai figurează încă pe lista problemelor nerezolvate.

Jocul de șah este totuși un joc finit. Aceasta înseamnă că, deși sunt enorm de multe mutări și variante, sau, dacă vreți, poziții distincte, numărul acestora este totuși limitat (finit). Imaginați-vă că aveți de numărat până la 10 miliarde. Dacă ați număra în fiecare secundă câte un număr ați avea nevoie de mai bine de trei sute de ani, și totuși sunt speranțe să duceți la capăt această problemă, chiar dacă pentru aceasta ar trebui să apelați la câteva generații care vă urmează. Zece miliarde este un număr finit. Problema este rezolvabilă și oricum, mai "domestică" decât, să zicem, cea a numărării până la... infinit, care este imposibilă.

Cititorul va scuza, poate, aceste digresiuni care n-au avut alt scop decât acela de a-l convinge asupra importanței și necesității rezolvării problemei de determinare a unei funcții de evaluare a mutărilor în șahul programat.

Ideea evaluării pe baza relațiilor existente între piesele și câmpurile tablei de șah, pe care autorul a prezentat-o în această lucrare, are la bază concepte cunoscute, precum cele de mobilitate a figurilor, de cooperare a pieselor, de atac, de apărare, de control a câmpurilor libere. Interdependența pieselor pe tabla de șah reprezintă, în concepția autorului, unicul criteriu serios de apreciere a unei poziții pe tabla de șah. Elementele de bază ale acestor interdependențe sunt relațiile de atac, de apărare și de control a câmpurilor libere. Așadar, în această accepțiune, nu valoarea intrinsecă a unei piese, de exemple a unui turn sau a unui nebun contează, ci totalitatea legăturilor acestor piese cu celelalte cu care intră în contact (relație) direct pe tabla de joc. Tot așa cum nu un câmp sau mai multe câmpuri libere, necontrolate de nici o piesă contează, ci doar cele care sunt la discreția pieselor de pe tablă. Evaluarea numerică a tuturor acestora, în contextul unei poziții, aceasta ar putea fi cheia unei funcții de evaluare eficiente.

Fără îndoială că scurtele lecții prezentate aici nu au putut să acopere toate problemele care alcătuiesc arsenalul de idei și concepte ale șahului-computer. Există numeroase detalii pe care programatorul trebuie să le aibă în vedere atunci când preocupările sale sunt îndreptate către un asemenea obiectiv. Autorul nutrește însă speranța că principalele noțiuni și concepte cuprinse în această lucrare ar putea fi de folos celor interesați în acest domeniu, și cu siguranță celor neinițiați. Cititorul poate alege, din conceptele prezentate, doar pe acelea care îl interesează. De asemenea, el poate să adapteze multe dintre acestea conform cu ideile și concepțiile proprii.

Dacă aici s-a preferat să se descrie reprezentarea tablei de șah în calculator sub forma matriceală, aceasta s-a făcut tocmai pentru că notațiile binecunoscute ale matricelor sunt și cele mai facile pentru o expunere și o înțelegere intuitivă simplă. Un programator va ști însă că adevărata reprezentare în calculator a unei table de șah, sau, dacă vreți, a unei matrice, se face printr-un lanț neîntrerupt de locații de memorie și pentru acces la o anumită informație, de exemplu, valoarea (conținutului) unui câmp al tablei de șah, este necesar să se cunoască adresa locației de memorie care o conține. Aceleași chestiuni sunt valabile și pentru alte date legate de problema programului de șah.

Reprezentarea tablei de șah sub forma matriceală poate fi utilă în mod deosebit celor care programează în limbaje de nivel înalt sau foarte înalt, unde modul de lucru cu tablouri de date este nu numai permis și detaliat, dar și foarte ușor. Desigur, în acest caz, este de așteptat ca performanțele programului, în ceea ce privește timpul de răspuns, să fie ușor mai scăzute.

Revenind asupra funcției de evaluare, chiar dacă ne repetăm, o facem

convinși că aceasta este adevărata problemă a unui program de șah. Ea este inima unui program de șah și, dacă ar fi să-l parafrazăm pe Goethe care consideră că “șahul este piatra de încercare a inteligenței umane”, s-ar putea spune că funcția de evaluarea din șahul-computer reprezintă “piatra de încercare a inteligenței artificiale”.

ASPECTE GEOMETRICE ALE TABLEI DE ȘAH

În cele ce urmează ne propunem să analizăm câteva aspecte geometrice ale tablei de șah cu referire la jocul de final. De ce finalurile? Deși ele sunt cunoscute ca fiind mai sărace și mai puțin spectaculoase, totuși abia aici se impune cu precădere elementul logic. În finalul jocului de șah numărul posibilităților imprevizibile este redus substanțial, dar această parte de joc rămâne deosebit de interesantă, dovadă în acest sens aducând-o, de exemplu, numeroasele studii artistice.

Despre finaluri se spune că mintea omenească, deși stăpână autoritară a tuturor legilor care domină această parte a jocului, nu le-a elucidat în întregime. Unele, chiar dintre cele elementare, mai sunt când și când puse în discuție și li se mai aduc uneori îmbunătățiri sau corecții.

De ce trebuiesc cunoscute finalurile? Simplu! Deoarece fără valorificarea avantajului material obținut sau fără lupta pentru salvarea partidei când avantajul este de partea adversarului jocul în sine ar fi lipsit de sens.

Din punct de vedere al planificării și programării, finalurile se împart în două mari categorii: cele în care există un avantaj material suficient pentru a obține victoria, și cele în care există un avantaj material sau pozițional de partea adversarului iar lupta se dă pentru salvarea partidei.

În teoria și practica șahistă se întâlnesc deseori reguli prestabilite care conduc la rezolvarea unică a unor poziții concrete. Majoritatea lor nu au un caracter universal, adică nu sunt valabile de la începutul și până la sfârșitul jocului, însă există și astfel de reguli. Ele țin, în principal, de mișcările particulare ale pieselor pe tabla de șah, deci de reguli elementare de manevrare.

Există însă un element de care se ține cel mai puțin seama în stabilirea unei strategii de joc. Acest element, de altfel foarte important, vizează *aspectele geometrice ale tablei de șah*.

La prima vedere tabla de șah pare a fi un pătrat împărțit în 64 de pătrățele mai mici, numite *câmpuri*. Nu insistăm asupra notației obișnuite a acestora care, până la cel mult o transformare liniară de coordonate, se identifică aproape perfect cu notația matriceală cunoscută în matematică, sau chiar cu cea de la unele rebusuri. Presupunem toate acestea, precum și problemele legate de regulile de desfășurare și convențiile de finalitate ale jocului, cunoscute.

Cât de inofensivă este această tablă de șah?

Dacă ne-am imagina jocul pe o tablă nemărginită, cu un număr nelimitat de câmpuri, având la bază același regulament de desfășurare, atunci aproape nimic din acest regulament nu trebuie schimbat, în afară poate de transformarea pionilor în figuri. Și totuși, jocul în acest caz devine evident imposibil în privința finalității lui. Sunt de neimaginat, de pildă, în astfel de situații, mături elementare cu turn și rege contra

rege, damă și rege contra rege, doi nebuni și rege contra rege etc. Mai mult, s-ar putea demonstra în astfel de situații că o figură, de exemplu un nebun, nu poate fi capturată niciodată.

Chiar și pe o tablă cu $16 \times 16 = 256$ câmpuri apar dificultăți mari de finalizare, majoritatea strategiilor de joc trebuind să fie modificate, iar unele, chiar abandonate.

Devine astfel evident faptul că în șah marginile tablei joacă un rol foarte important.

Majoritatea finalurilor elementare au la bază o geometrie a matului în care regele slab este dus obligatoriu la marginea tablei. Mai mult, există unele finaluri care nu pot fi concepute fără constrângerea regelui slab într-unul din colțurile tablei de șah. Astfel de mături, cu participarea marginilor tablei, presupun, desigur, existența unui avantaj material suficient, de regulă, cel puțin de valoarea unui turn.

Foarte rar sunt posibile mături în care regele slab este încolțit definitiv pe un câmp din mijlocul tablei, deci care se lipsesc de aportul concret al marginilor. Astfel de mături pot fi realizate doar cu un avantaj material suficient de mare (mult mai mare decât în celelalte cazuri), avantaj care poate fi invers proporțional cu aportul, conștient sau nu, al pieselor regelui slab.

Analizând această ultimă categorie de mături se constată că, față de situația precedentă, diferența de material necesară trebuie să fie suficientă pentru delimitarea unei margini unde regele slab va fi constrâns. Într-adevăr, un singur turn, de exemplu, poate delimita o margine pentru regele slab (o linie sau o coloană), alta decât cea a tablei de șah, după care fizionomia finalului devine aceeași ca în cazul precedent. Deci, o primă concluzie ar fi aceasta: *dacă n-avem margini apropiate, trebuie să le construim cu propriile noastre figuri*. Tot la fel și cu colțurile.

O astfel de strategie nu este una particulară, ci delimitează reguli cu caracter general ce trebuiesc urmărite și aplicate pe tot timpul desfășurării unui final de partidă.

Diferența de material despre care vorbeam mai înainte se referă, desigur, la piesele active (cele care pot interveni imediat în luptă) pe tabla de șah. Jucătorul condus de aceste reguli va fi întotdeauna mai puternic. Nu vom insista însă asupra acestor chestiuni, ci vom merge mai departe să căutăm și alte principii generale ale jocului, cu rol preponderent în rezolvarea finalurilor.

Se spune în șah, și nu numai, că este mai ușor să te aperi decât să ataci. Mulți șahisti adoptă jocul pozițional care, desigur, implică mai puțină participare și bătaie de cap, nu însă și mai puțină dificultate. De multe ori apar situații cu diferențe strategice sau materiale care trebuiesc recuperate. În astfel de cazuri, cel mai bun plan de joc vizează lupta pentru echilibrarea situației care, dacă este vorba despre un final de partidă, ar trebui să se concretizeze în remiză.

Geometria tablei de șah naște, în astfel de situații, probleme paradoxale. Conjugarea pe o cale de mijloc a mai multor strategii, atunci când acestea, luate separat, nu duc la nimic, poate deseori rezolva probleme care sfidează chiar limita bunului simț.

Un exemplu concludent în acest sens a fost pus în evidență printr-un studiu celebru al reputatului șahist-matematician Richard Reti, în anul 1921. Tema studiului este "albul mută și face remiză" iar poziția prezentată în diagrama nr.1, conține la *alb*: Rh8 și c6, iar la *negru*: Ra6 și h5.

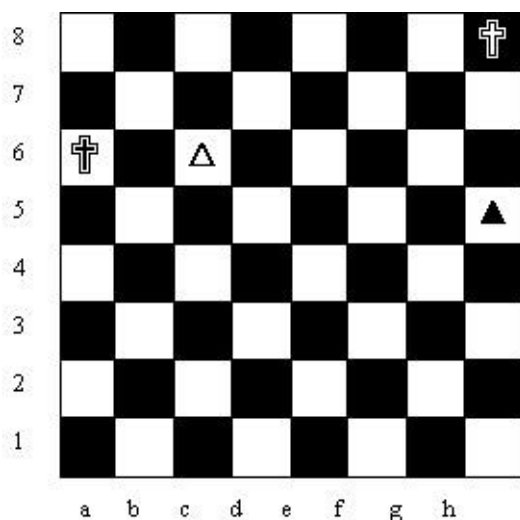


Diagrama nr. 1
Albul mută și face remiză (Reti-1921)

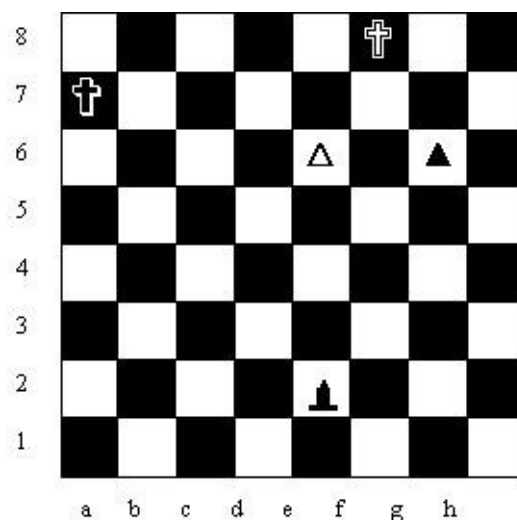


Diagrama nr. 2
Albul mută și face remiză (Reti-1929)

La prima vedere poziția pare disperată pentru alb, deoarece pionul negru are două tempouri în plus spre linia de transformare și, în același timp, nu se vede nici o șansă de promovare a propriului său pion care este și prea îndepărtat de regele său și poate fi capturat ușor de regele advers. Există trei planuri de joc pentru alb, toate părând la fel de nesatisfăcătoare. Să le analizăm pe rând:

P1. Alergarea cu regele pentru a prinde pionul negru. Cel mai scurt drum pare a fi coloana h, deci 1. Rh7, 2. Rh6, 3. Rh5 etc., însă este clar că acest pion nu poate fi prins. De altfel, în asemenea cazuri există și o regulă care clarifică rapid situația, numită "regula pătratului". Aceasta afirmă că: *dacă regele se află sau, fiind la mutare, poate să pătrundă, în pătratul mișcător al pionului care se îndreaptă spre linia de transformare, pătrat imaginar cu latura egală cu numărul câmpurilor pe care le are de parcurs pionul spre transformare, două dintre laturile acestui pătrat fiind linia de transformare și cea pe care se află pionul, atunci pionul poate fi ajuns și capturat. În caz contrar, acesta poate ajunge liniștit pe ultima linie, unde se va transforma într-o figură.* O regulă geometrică și nimic mai mult, care surprinde prin simplitatea și utilitatea ei!

În cazul problemei noastre, albul fiind mutare, nu poate pătrunde cu regele în pătratul pionului constituit din vârfurile h5, h1, d1 și d5, deci planul pare absurd.

P2: Încercarea de a sări în sprijinul propriului său pion prin deplasarea regelui pe cel mai scurt drum până la acesta. Și în acest caz se vede că planul este total nesatisfăcător, întrucât regele advers poate captura pionul alb cu mult înainte ca regele său să-l poată sprijini.

P3: Încercarea de promovare a propriului pion prin avansare imediată. Însă, conform regulii pătratului, regele aflându-se deja în pătratul pionului, poate să-l ajungă și să-l captureze.

Atunci, ce este de făcut? Trebuie abandonată lupta? Încă nu! Mai există un plan, desigur, mai greu de intuit, care are la bază o soluție de compromis: conjugarea celor trei strategii particulare într-una singură. Altfel spus, determinarea unui drum optim (variantă) care să aibă în vedere abordarea tuturor variantelor particulare pe o cale de mijloc. Jucând 1. Rg7, la prima vedere se pare că albul nu și-a îmbunătățit cu nimic situația, însă ceva tot s-a realizat: printr-o singură mutare el a făcut câte un pas

în fiecare dintre planurile P1 și P2, cel de-al treilea plan rămânând în continuare posibil de urmat. Acum, oricare ar fi răspunsul negrului situația albului rămâne îmbunătățită cel puțin cu un pas.

Cel mai bun răspuns al negrului pare a fi 1... h4. Continuând raționamentul în spiritul aceleiași idei generale de joc albul ar trebui să joace 2. Rf6 după care apar imediat și unele speranțe pentru sprijinirea pionului său, deoarece la 2... h3 se poate juca 3. Re6 h2 4. c7 Rb7, 5. Rd7 și remiză. De aceea, negrul este nevoit să joace 2... Rb6 pentru a preîntâmpina amenințarea însă, în spiritul aceleiași idei care domină acum jocul albului, se va juca 3. Re5!!

Abia acum lucrurile par să se fi lămurit pe deplin întrucât, dacă negrul continuă cursa de transformare, albul ajunge la timp să-și sprijine pionul, iar dacă negrul bate pionul alb, regele alb poate pătrunde în pătratul de transformare al pionului negru (de exemplu, 4. Rf4) și ca urmare îl va putea captura la timp.

Astfel, situația paradoxală de la început a putut fi depășită prin aplicarea unei strategii corespunzătoare de joc. Nu însă o strategie oarecare, ci una generală, care a cuprins toate planurile particulare de joc existente în poziția concretă de pe tabla de șah. De ce a fost posibilă această strategie? Explicația logică este strâns legată de mișcarea particulară a pieselor pe tabla de șah. În cazul problemei noastre mișcarea regelui s-a făcut după o geometrie specială, în care *diagonala pătratului este egală cu latura*.

Observația are un caracter universal și orice strategie de joc (cu atât mai mult cele programate) trebuie să țină seama de acest aspect.

Un studiu asemănător, având aceeași temă, la fel de celebru și aparținând aceluiași neobosit cercetător Reti, este cel prezentat în diagrama nr. 2, cu poziția la alb: Rf8, e6, și la negru: Ra7, Ne2, g6. Se afirmă despre această poziție că și un campion mondial dacă ar fi întâlnit-o în timpul unei partide, ar fi cedat-o, într-atât pare de dezastruoasă situația pentru alb. Și totuși, poziția poate fi salvată, chiar și de un șahist mediocre, dacă este înarmat cu raționamente de genul celor expuse mai sus. Planurile particulare de joc pentru alb sunt:

P1: Pătrunderea cu regele în pătratul de transformare al pionului negru

P2: Anihilarea nebunului negru care, prin manevra b5 poate împiedica transformarea pe e8 a pionului alb

P3: promovarea pionului alb.

Care este atunci strategia generală de joc, care îmbunătățește la fiecare pas situația albului? Lăsăm cititorul să reflecteze asupra acesteia, expunându-i doar soluția problemei: 1. Re7 g5 2. d6! g4 3. 7 Nb5 4. Rc5!! și, după mutarea nebunului oriunde, urmează 5. Rd4 g3 6. Re3 și remiză. O cursă diabolică în zigzag, sfidând parcă orice logică a bunului simț. O cursă care, aparent, nu respectă legea celei mai scurte distanțe dintre două puncte. Mult mister, dar și multă satisfacție. Întocmai cum spunea marele gânditor N. Iorga: "*Șahul este cel mai minunat mijloc de disciplinare a gândirii, de ordonare a domeniilor de cunoaștere și de recreație a spiritului. Este un univers comprimat pe 64 de pătrățele*".

Există și alte legi geometrice ale tablei de șah? Cu siguranță. Sarcina unui jucător este să descifreze acele reguli care domină jocul și să le aplice în situații concrete. Iar sarcina unui programator de șah este aceeași, iar în plus trebuie să modeleze matematic aceste reguli și să le transforme în limbajul calculatorului.

Unul din cele mai importante principii, valabil în finalurile de regi și pioni, este cel al *opoziției*. Fostul mare campion al lumii, J. L. Capablanca afirma în legătură cu aceasta: "*opoziția și legile ei pot fi abordate într-o manieră pur matematică*".

Este tocmai ceea ce trebuie să facă un programator care dorește să modeleze matematic aceste categorii de reguli. Și ele sunt multe. Mai amintim doar câteva dintre ele, lăsând cititorul să reflecteze asupra modului în care acestea pot fi

implementate: pionul care blochează doi pioni, avansarea rapidă a pionilor începând cu pionul care nu întâlnește alt pion în calea sa, ocuparea rapidă a centrului cu regele în finalurile de pioni, plasarea regelui în fața pionilor care avansează etc.

Subiectul analizat este larg și nu ne-am propus să-l epuizăm aici.

În final mai facem doar o singură observație generală privind planificarea și programarea finalurilor. Am văzut că în acele finaluri în care lupta se dă pentru salvare, o modalitate de abordare algoritmică a acestora se bazează pe determinarea unei variante care să îmbunătățească poziția la fiecare mutare cu câți mai mulți pași în cât mai multe strategii particulare simultan.

Pentru cealaltă categorie de finaluri, în care avantajul material este suficient pentru a aborda matul, pot fi avute în vedere următoarele aspecte: a) majoritatea finalurilor elementare necesită un avantaj material de cel puțin valoarea unui turn pentru a realiza matul, cu ajutorul marginilor sau colțurilor tablei b) toate finalurile de acest gen au la bază constrângerea regelui slab, reducându-i spațiul de mișcare până la mutarea decisivă c) orice mat în mijlocul tablei se poate reduce în principiu la cel descris la punctul a), diferența constând în necesitatea unui avantaj material mai mare, suficient pentru restrângerea spațiului de mișcare a regelui slab b).

Pe baza acestor observații se poate constata că o strategie care să aibă la bază reducerea spațiului de mișcare a regelui slab, corelată cu determinarea mutării decisive este tocmai adecvată pentru abordarea programată a acestor categorii de finaluri.

FUNCȚIA DE EVALUARE ÎN ȘAHUL PROGRAMAT

Cea mai dificilă problemă întâmpinată de programatorii de șah este cea legată de aprecierea unei poziții care se concretizează de regulă prin alegerea unei mutări.

După cum se știe această problemă nu este deloc simplă nici chiar pentru un jucător uman. Jucătorul de șah, indiferent câte semimutări anticipează pe tabla de șah, se oprește în cele din urmă – la capătul variantelor generate – asupra unei poziții rezultate și o analizează, căutând să răspundă cât mai corect la întrebări de genul: “Stau mai bine?”, “Stau rău?”, “Este o poziție de egalitate?” ș.a.

De asemenea, pe parcursul generării variantelor, jucătorul nu ia în seamă toate mutările posibile, ci doar pe acelea care, din considerente proprii, par a fi cele mai bune.

Tot la fel ar trebui să procedeze și calculatorul, însă el, așa cum se știe, “judecă” doar prin “cifre”. Apare astfel ideea utilizării unei funcții de evaluare numerică, funcție care, din punct de vedere matematic, în practică i se dă o formă lineară, care presupune un număr restrâns de calcule, având la bază operații elementare, de regulă, cele de adunare și de înmulțire. Dar forma lineară aleasă poate fi considerată ca fiind abuzivă căci, în realitate, numai puține elemente care intră în componența ei au comportări liniare.

Oricare ar fi o funcție de evaluare, ea trebuie să înglobeze obligatoriu criterii precum: diferența de material, controlul centrului, structura pionilor, mobilitatea figurilor, siguranța regelui, posibilități de atac, de apărare etc.

A stabili, de exemplu, o dependență liniară între controlul centrului și mobilitatea anumitor figuri este totuși un abuz, care nu se justifică suficient prin câștigul de timp pentru calcul și evaluare.

În prezent, majoritatea programelor cunoscute folosesc funcții asemănătoare de evaluare, care diferă doar prin numărul de termeni și valorile particulare atribuite parametrilor. O funcție de evaluare ideală încă nu a fost descoperită, însă un lucru este cert: exprimarea ei lineară este o arproximare grosieră, cu atât mai evident cu cât avem de-a face cu un domeniu discret de valori.

De unde provin aceste dificultăți? În primul rând de la numărul relativ mare de parametri criteriali luați în calcul; de exemplu, pentru calculul parametrului privind posibilitățile de atac sunt necesare estimări ale altor parametri ajutători, cum ar fi: piese avansate în tabăra adversă, turnuri plasate pe coloane deschise sau semideschise, controlul câmpurilor vecine regelui, atacuri la rege, legarea pieselor, atacul dublu sau combinat etc.

În al doilea rând, dificultăți mari sunt întâmpinate și la determinarea coeficienților de corelație liniară a parametrilor criteriali. Acești coeficienți, și ei mulți la număr, sunt determinați pe căi statistice sau stochastice.

Una din căile sigure de elaborare a unei funcții de evaluare cât mai exacte

constă în reducerea numărului de parametri utilizați.

Părăsim pentru moment problema funcției de evaluare și aducem în discuție următoarea chestiune: ce este valoarea dinamică a unei piese pe tabla de șah și ce rol joacă ea? După cum se știe, în orice problemă de șah întâlnim diferențe dintre valoarea intrinsecă a unei figuri și utilitatea ei la un moment dat. De exemplu, un turn valorează, după unele criterii statice, aproximativ cât cinci pioni, însă această valoare nu se menține pe tot parcursul partidei. Sunt situații (poziții) în care rolul acestui turn crește atât de mult, încât poate depăși chiar valoarea statică a celei mai puternice figuri. Dar sunt și cazuri în care valoarea acestuia poate să scadă (de exemplu, situații “out of game”).

Ce anume influențează această valoare? Evident, ea se modifică în raport cu participarea piesei la joc, adică există o dependență semnificativă între această piesă și toate celelalte cu care ea intră în contact direct, prin relații de atac și de apărare, dependență care, caracterizată valoric, determină așa numita valoare dinamică a piesei respective. Se nasc astfel întrebări de genul: cât valorează de fapt un turn care atacă un nebun, este atacat de un cal, apără un cal și un pion și controlează nouă câmpuri libere? (în toate fazele jocului controlul câmpurilor libere joacă un rol foarte important). Aceste probleme inexacte sugerează ideea unui joc în care strategia de bază este acțiunea coordonată și cooperarea armonioasă a pieselor.

Despre aceasta, fostul campion mondial J. R. Capablanca, într-una din cărțile sale de șah, a făcut observația că semnifică “principiul de bază care guvernează întreaga partidă”. Acest principiu se concretizează pe tabla de șah prin mobilizarea figurilor și activizarea lor, în așa fel ca prin acțiunea lor comună să ridice cât mai multe probleme adversarului.

O piesă dintr-o poziție oarecare este caracterizată de următoarele categorii de relații:

- a. se află în relația de apărare cu piese proprii (apără și/sau este apărată)
- b. se află în relația de atac cu piese adverse (atacă și/sau este atacată)
- c. controlează unul sau mai multe câmpuri libere.

Unele dintre relații sunt favorabile taberei din care face parte piesa, altele nu.

Vom face acum observația importantă că strategia relațională despre care vorbim are avantajul că acoperă aproape în întregime sistemul criterial care stă la baza unei funcții de evaluare. Astfel, o piesă va fi caracterizată potențial mai bine atunci când va deveni furnizoarea mai multor relații favorabile. Aceasta nu implică altceva decât un joc mai activ, cu o mobilitate mai mare a pieselor și, ca o consecință imediată, cu oportunități de soluționare a unor probleme precum:

- centralizarea pieselor,
- controlul centrului (care conferă figurilor fronturi mai largi de acțiune),
- atacul unor zone importante,
- ocuparea cu turnul a unei coloane deschise (și asta nu întotdeauna, așa cum s-ar întâmpla într-o strategie dispartă, ci numai atunci când vechiul obiectiv al turnului poate fi părăsit),
- cedarea unei figuri slabe pentru una foarte activă a adversarului (nu înseamnă asta, deseori, un început de combinație?) etc.

Mulțimea celor trei tipuri de relații existente la un moment dat într-o poziție poate defini complet poziția. Mai mult, modalitatea de descriere a pozițiilor pe baza relațiilor are și avantajul că grupează poziții care sunt considerate a fi distincte în sistemul clasic, în clase mai largi de poziții, conform caracteristicilor relaționale comune.

O mutare regulamentară efectuată pe tabla de șah transformă poziția P_0 în poziția P_1 . Din punct de vedere relațional, ne punem întrebarea prin ce diferă cele

două poziții? Răspunsul este ușor de întrevăzut: orice mutare efectuată anulează câteva relații existente și, în același timp, introduce câteva relații noi în poziția P_0 , transformând-o în P_1 . De regulă, marea majoritate a relațiilor vechi nu sunt afectate de o mutare, rămânând intacte. Așa cum se poate stabili statistic, relațiile care sunt afectate în urma unei mutări reprezintă circa 5% din totalul relațiilor care caracterizează poziția.

Dacă vom evalua numeric toate aceste relații, vom observa, fapt deosebit de important, că valorile acestora pot fi niște constante care, pentru a fi determinate, se pot utiliza diferite metode statistice.

Însumarea algebrică a acestor valori, deci exprimarea cantitativă a relațiilor dinamice, reprezintă un criteriu calitativ de departajare a mutărilor.

Pentru corelarea celor trei tipuri de parametri relaționali (de apărare, de atac și de control) pot fi folosiți trei coeficienți multiplicativi reali, pozitivi și subunitari, a căror sumă este egală cu unitatea și care transformă funcția de evaluare într-o combinație lineară de funcții criteriale. Acești coeficienți pot fi și ei determinați statistic, de exemplu, analizând și caracterizând numeric diferențele poziționale (de preferință, preluate din partidele marilor maeștri), care au survenit după mutări care au răsturnat un echilibru ce părea evident.

În acest mod, rezolvând aceste probleme pur matematic, vom avea dovada unei funcții de evaluare care, deși are baze empirice, este totuși mai simplă și adesea poate fi îmbunătățit în mod experimental.